待ち行列システムの最適制御に関する計算アルゴリズムについて (待ち行列理論とその応用 II)

大野, 勝久

数理解析研究所講究録
452巻 1982年 1-19

待ち行列システムの最適制御に関する計算アルゴリズムについて

京大 工学部
大野勝久

1. はじめに

待ち行列システムの最適制御における主要な問題は，その最適制御政策のもつ性質を明らかにし，その政策を効率的に決定することである．しかし従来の研究は最適政策の状態に関する単調性等を示すにとどまり［1－5］，示された性質を有効に利用した最適政策の決定アルゴリズムまで論じたものは［6，7］があるにすぎない。また，単調性等が示された問題も，M／G／1，GI／G／1，M／M／2 システムにおけるサーバー数の制御問題，M／M／1 システムにおけるサービス率の制御問題等簡単なものに限られている。

待ち行列システムの最適制御問題は通常セミ・マルコフ決定過程として定式化される。有限状態，有限決定のセミ・マルコフ決定過程にたいしては最適政策が定常になることが保証され，それを求めるアルゴリズムとして，

（i）　政策反復法

（ii）　逐次近似法

（iii）　線形計画法

（iV）　修正政策反復法

が知られている．したがって最適定常政策のもつ性質を何んら知らなくても，これらアルゴリズムを用いれば最適定常政策を容易に決定できるように思われるかもしれない。しかし待ち行列システムの最適制御問題は簡単な問題でさえ状態数が数百以上となり，（i）－（iii）のアルゴリズムでは容易に解けない問題になる。本報告では，M／M／C システムおよびC段直列M／M型待ち行列システムにおけるサーバー数の制御をも含むサービス率の制御問題を考え，その最適定常政策を求める修正政策反復法［８］によるアルゴリズムを提案し，数値例によりその実用可能性を検討する。

２．M／M／C待ち行列システムの最適制御

図１に示されるM／M／C待ち行列システムを考える．客は平均到着率λのポアソン過程に従って到着し，行列長はNに制限されているものとする。問題は，サーバーi（$i=1,\cdots,c$）のとりうるサービス率が

$$\{0,\ \mu_i,\ \cdots,\ M_i\mu_i\}$$

で与えられるとき，システムの状態に応じて各サーバーのサ

ービス率をどのように制御すればよいかを求めることである。それに伴い，遊休中（サービス率が正でかつ客をサービスしていない）のサーバーが複数個存在する時，どのサーバーに客を割りあてるかも制御される。特に，全てのiで $M_i=1$, $\mu_i=\mu$ とおけば，この問題はM／M／Cにおけるサーバー数の制御問題になる。

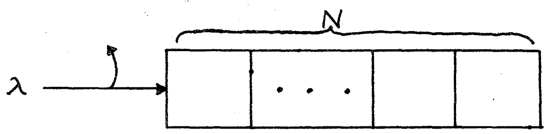


図 1

サーバー	サービス率
①	$\{0, \mu_1, \cdots, M_1\mu_1\}$
⋮	
ⓘ	$\{0, \mu_i, \cdots, M_i\mu_i\}$
⋮	
ⓒ	$\{0, \mu_c, \cdots, M_c\mu_c\}$

このシステムの状態は客の到着あるいはサービスの終了時点でのみ変化し，したがってサービス率の制御およびサーバーへの客の割り当てもまた，客の到着あるいはサービスの終了直後に行なわれる。以下，自然な仮定として

（C 1） 休止中（サービス率が0）のサーバーへ客を割り当てることはない。

（C 2） 遊休中（サービス率が正でかつサービスをしていない）のサーバーは，待っている客がいる限り客を割り当てられる。

（C 3） 稼動中（現在サービス中）のサーバーのサービス

率は変えられず，サービス中の客を追いだすこともない。

をおき，費用として

h_1 ： 待ち行列中の客の１人当り単位時間当り待ち費用

h_2 ： サービス中の客の１人当り単位時間当り待ち費用

W_{1i} ： サーバーiの単位サービス率，単位時間当り稼動費用

W_{2i} ： サーバーiの単位サービス率，単位時間当り遊休費用

d_i ： サーバーiの単位サービス率　当り切り換え費用

e_i ： サーバーiの単位サービス率の切り換えに要する固定費用

を考える。

α： 割引率 （$\alpha > 0$）

とおく。問題は，無限期間にわたる平均割引費用を最小にするサービス率および客の割り当てを与える最適定常政策を求めることである。

この問題をセミ・マルコフ決定過程として定式化する。まず客の到着あるいはサービスの終了直後におけるシステムの状態Sを

$$S = (X,\ Y_1, \cdots, Y_c,\ Z_1, \cdots, Z_c) \tag{2.1}$$

で表わす．ここで

$X \in \{0,\ 1,\ \cdots,\ N\}$ ： 待ち行列長

$Y_i \in \{0,\ 1,\ \cdots,\ M_i\}$ ： （サーバーiの現在のサービス率）$/\mu_i$

$$Z_i = \begin{cases} 0, & \text{サーバーiが遊休あるいは休止中} \\ 1, & \text{サーバーiが稼動中} \end{cases} \quad (i=1,\cdots,c)$$

である．$\bar{S}$で可能な状態Sの集合を表わすことにすれば，$(0, 0, \cdots, 0)$等の状態は起こりえないから，$\bar{S}$の要素数はたかだか

$$2^c (N+1) \prod_{i=1}^{c} (M_i + 1) \tag{2.2}$$

である。システムが客の到着あるいはサービスの終了で状態Sに遷移したとき，その状態Sをみて決定の集合A(S)から決定aが選ばれる。決定aは

$$a = (Y'_1, \cdots, Y'_c,\ Z'_1, \cdots, Z'_c) \tag{2.3}$$

と表わされる．ここで

$Y'_i \in \{0, 1, \cdots, M_i\}$ ： （サーバーiのサービス率）$/\mu_i$

$$Z'_i = \begin{cases} 0, & \text{サーバーiに客を割り当てない．} \\ 1, & \text{サーバーiに客を割り当てる。} \end{cases} \quad (i=1,\cdots,c)$$

である．仮定 C1) ～ C3) から各状態Sで可能な決定の集合A(S)が定められる。たとえば

$$A(1,2,1,\cdots 1,\overset{z_1}{0},1,\cdots,1)=\{(0,1,\cdots,1,0,1,\cdots,1),(1,1,\cdots,1,1,1,\cdots,1),$$
$$\cdots,(M_1,1,\cdots,1,1,1,\cdots,1)\}$$

である。特に，$\mu_1=\mu_2$，$M_1=M_2\cdots$，$e_1=e_2$のようにサーバー1，2が同じである場合，状態$S=(1,0,0,\cdots,\overset{z_1}{0},\overset{z_2}{0},\cdots)$における決定$a_1=(\overset{Y_1}{1},\overset{Y_2}{0},\cdots,\overset{z_1}{1},\overset{z_2}{0},\cdots)$と$a_2=(\overset{Y_1}{0},\overset{Y_2}{1},\cdots,\overset{z_1}{0},\overset{z_2}{1},\cdots)$は本質的に同じ決定になり，このような決定のうちの1つa_2は$A(1,0,0,\cdots,0,0,\cdots)$から除外される。

状態$S=(X,Y_1,\cdots,Y_c,Z_1,\cdots,Z_c)$で決定$a=(Y_1',\cdots,Y_c',Z_1',\cdots,Z_c')$をとると状態は瞬間的に中間状態

(2，4) $$S'=(X',Y_1',\cdots,Y_c',Z_1',\cdots,Z_c')$$

へ移る。ここでY_i'，Z_i'は決定aにより定まる次の遷移までのサーバーiのサービス率と状態を示し，X'はaによって変化した待ち行列長であり

(2，5) $$X'=X+\sum_{i=1}^{c}(Z_i-Z_i')$$

で与えられる．システムは次の客の到着あるいはサービス終了直前までこの状態S'にとどまり，客が到着すれば

(2，6) $$S_0=(X'+\delta(X'),Y_1',\cdots,Y_c',Z_1',\cdots,Z_c')$$

へ遷移し，サーバーiのサービスが終了すれば

(2，7) $$S_i=(X',Y_1',\cdots,Y_c',Z_1',\cdots,\overset{i}{0},\cdots,Z_c')$$

へ遷移する．以上の状態遷移は図2のように図示される．

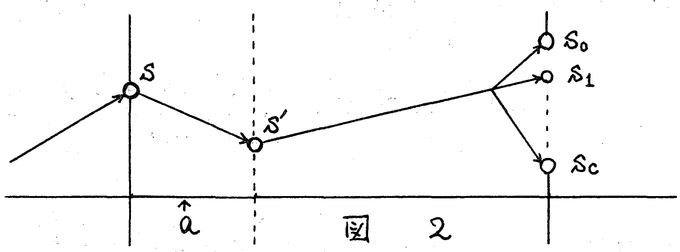


図 2

ここで，$\delta(X)$は

(2, 8) $$\delta(X)=\begin{cases}0, & X=N\text{のとき}\\ 1, & X\neq N\text{のとき}\end{cases}$$

である。 状態Sで決定aをとったとき状態S_0へ遷移する割引遷移確率 $q_{ss_0}(a)$ は

(2, 9) $$q_{ss_0}(a)=\int_0^\infty e^{-\alpha t}\lambda\delta(X')\exp\{-\lambda\delta(X')t\}\exp\{-\sum_{i=1}^{c}Y_i'Z_i'\mu_i t\}\,dt$$
$$=\frac{\lambda\delta(X')}{\lambda\delta(X')+\sum_{i=1}^{c}Y_i'Z_i'\mu_i+\alpha}$$

で与えられる。同様にS_iへの割引遷移確率$q_{ss_i}(a)$は

(2, 10) $$q_{ss_i}(a)=\frac{Y_i'Z_i'\mu_i}{\lambda\delta(X')+\sum_{i=1}^{c}Y_i'Z_i'\mu_i+\alpha}\qquad(i=1,\cdots,c)$$

である。また，状態Sで決定aをとったとき次の遷移までにかかる平均割引費用 $r(S,a)$は，$\varepsilon(Y)$を

(2, 11) $$\varepsilon(Y)=\begin{cases}0, & Y=0\text{のとき}\\ 1, & Y\neq 0\text{のとき}\end{cases}$$

で与えれば

$$(2,12)\quad r(s,a)=\sum_{i=1}^{c}\{\mu_i d_i\,|Y_i-Y_i'|+e_i\varepsilon(Y_i-Y_i')\}$$
$$+\frac{h_1X'+\sum_{i=1}^{c}\{h_2 Z_2'+\mu_i Y_i'(w_{1i}Z_i'+w_{2i}(1-Z_i'))\}}{\lambda\delta(X')+\sum_{i=1}^{c}Y_i'Z_i'\mu_i+\alpha}$$

となる。したがってM/M/Cの最適制御問題は $s\in\bar{S}$ にたいして

$$(2,13)\quad v^*(s)=\min_{a\in A(s)}\{r(s,a)+\sum_{j=0}^{c}q_{ss_j}(a)v^*(s_j)\}$$
$$=r(s,f^*(s))+\sum_{j=0}^{c}q_{ss_j}(f^*(s))v^*(s_j)$$

をみたす最小費用 $v^*(s)$ および最適定常政策 $f^*(s)$ を求めるセミ・マルコフ決定過程に帰着された。

3. 修正政策反復法

前節で定式化された問題を解く修正政策反復法を示す。

$$(3.1)\quad \alpha(s,a)=\sum_{j=0}^{c}q_{ss_j}(a)=(\lambda\delta(X')+\sum_{i=1}^{c}\mu_iY_i'Z_i)/(\lambda\delta(X')+\sum_{i=1}^{c}\mu_iY_i'Z_i'+\alpha)$$

$$(3.2)\quad \beta=\max_{s,a}\{\alpha(s,a)\}=(\lambda+\sum_{i=1}^{c}\mu_iM_i)/(\lambda+\sum_{i=1}^{c}\mu_iM_i+\alpha)$$

$$(3.3)\quad \gamma=\min_{s,a}\{\alpha(s,a)\}=\min\{\lambda,\mu_1,\cdots,\mu_c\}/\{\min\{\lambda,\mu_1,\cdots,\mu_c\}+\alpha\}$$

とおく

スラップ0： $C_0=\max_{s\in S}\min_{a\in A(s)}\{r(s,a)\}/(1-\beta)$ を計算す

る。

ステップ1： 非負整数 m，正定数 ε，十分大きな正数 M を定め，$s \in \bar{S}$ にたいして

$$u^0(s) = M, \quad v^0(s) = C_0, \quad A^0(s) = A(s)$$

とおき，$\eta_0 = 0$，$n = 1$ とおく。

ステップ2： $s \in \bar{S}$ にたいして

(3.4) $$w^n(s) = \min_{a \in A^{n-1}(s)} \left\{ r(s,a) + \sum_{j=0}^{c} q_{ss_j}(a) v^{n-1}(s_j) \right\}$$

を計算し，同時に

(3.5) $$A^n(s) = \left\{ a \in A^{n-1}(s) ; r(s,a) + \sum_{j=0}^{c} q_{ss_j}(a) v^{n-1}(s_j) \le u^{n-1}(s) - \alpha(s,a)\eta_{n-1} \right\}$$

を定める。$f^{n-1}(s)$ が $w^n(s)$ を与えれば $f^n(s) = f^{n-1}(s)$ とおき，さもなければ $f^n(s)$ を $w^n(s)$ を与える任意の決定にとる。もし，$A^n(s)$ が全ての s で唯一となればステップ5へ行く。（注意．$r(s.a)$，$q_{ss_j}(a)$，$\alpha(s,a)$ はメモリーに記憶せず，必要に応じて (2.9) −(2.12)，(3.1) から計算される．）

ステップ3： $y^0(s) = w^n(s)$ とおき，$l = 0, 1, \cdots, m-1$ にたいして

(3.6) $$y^{l+1}(s) = r(s, f^n(s)) + \sum_{j=0}^{c} q_{ss_j}(f^n(s)) y^l(s_j)$$

$(s \in S)$ を計算し，$v^n(s) = y^m(s)$ とおく。

ステップ4：

$$\begin{cases} \Delta_n = \max_s \{ v^n(s) - v^{n-1}(s) \}, & \nabla_n = \min_s \{ v^n(s) - v^{n-1}(s) \} \\ a_n = \max_s \{ w^n(s) - v^n(s) \}, & b_n = \min_s \{ w^n(s) - v^n(s) \} \end{cases} \tag{3.7}$$

$$\begin{cases} \xi_n = \min \{ -\gamma^m b_n / (1-\gamma^m),\ (\gamma \Delta_n + a_n)/(1-\gamma) \} \\ \eta_n = (\beta \nabla_n + b_n)/(1-\beta) \end{cases} \tag{3.8}$$

を計算する。もし

$$\xi_n - \eta_n \geq 2\varepsilon \tag{3.9}$$

ならば $u^n(s) = v^n(s) + \xi_n \quad (s \in \bar{S})$，

$n = n+1$ とおき，ステップ2へ戻る。さもなければ

$$v(s) = v^n(s) + (\xi_n + \eta_n)/2 \quad (s \in \bar{S}) \tag{3.10}$$

は $v^*(s)$ の ε 近似値である。

$$\begin{cases} \delta_n = \min_s \{ v(s) - r(s, f^n(s)) - \sum_{j=0}^{c} q_{ss_j}(f^n(s))\, v(s_j) \} \\ \varepsilon_n = \begin{cases} \varepsilon - \delta_n / (1 - \min_s \alpha(s, f^n(s))), & \delta_n \geq 0 \text{ の時} \\ \varepsilon - \delta_n / (1 - \max_s \alpha(s, f^n(s))), & \delta_n < 0 \text{ の時} \end{cases} \end{cases} \tag{3.11}$$

とおけば，f^n は ε_n-最適政策である。

ステップ5： $\{ f^n(s) ;\ s \in \bar{S} \}$ は唯一の最適定常政策である。

$$c_n = \max_s \{ w^n(s) - v^{n-1}(s) \}, \quad d_n = \min_s \{ w^n(s) - v^{n-1}(s) \} \tag{3.12}$$

を計算すれば，$v^*(s)$ は $s \in \bar{S}$ にたいして

$$v(s) = w^n(s) + \frac{1}{2} \{ \gamma c_n / (1-\gamma) + \beta d_n / (1-\beta) \} \tag{3.13}$$

で評価される。

上記アルゴリズムで記憶すべきものは，$v^{n-1}(s)$，$v^{n}(s)$，$w^{n}(s)$，$u^{n}(s)$，$A^{n}(s)$，$f^{n}(s)$ であり，必要な記憶容量は大体

$$5|\bar{S}| + \sum_{s \in \bar{S}} |A(s)| \qquad (3.14)$$

である。ここで $|\cdot|$ は・の要素数を表わす。

上記アルゴリズムでは v^{0} としてステップ0でえられた c_0 を用いたが，(3.8)式を一部修正すれば任意の v^{0} を用いることができる。また $m=0$ とすれば上記アルゴリズムは（ii）逐次近似法に一致し，$m \to \infty$ とすれば（i）政策反復法における値決定ルーチンを逐次近似法で解くアルゴリズムに一致する。詳細は［8］を参照されたい。

4. 数値例

$c=2$, $\mu_1=\mu_2=0.01$, $M_1=M_2=4$, $\lambda=0.06$, $N=10$

とおいた例を考える。サーバー1，2は同一であるから，たとえば状態（1，3，2，0，1）と（1，2，3，1，0）は区別できず，可能な状態数は 740 となる。また可能な決定の総数は 4980 となる。

この問題を（i）政策反復法で解くためには 740元連立一次方程式を解かねばならず，ガウスの消去法では相当困難で

ある。

(iii) 線形計画法で解くためには，4980変数740制約のLP問題を解かねばならず，やはり困難である。以下前節のアルゴリズムによる結果を示す。

(1) $h_1=10,\ h_2=0,\ w_{11}=w_{12}=500,\ w_{21}=w_{22}=250$

$d_1=d_2=5000,\quad e_1=e_2=0$

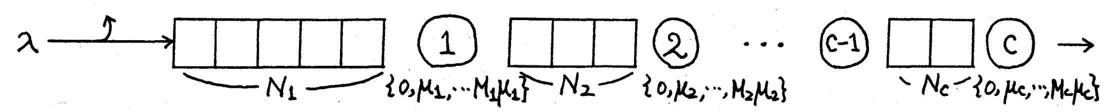
とした問題が $m=10,\ \varepsilon=0.01$ とおいた修正政策反復法で約19秒で解かれ，唯一の最適定常政策（ステップ5）がえられた。[9]のアルゴリズムで状態を整理した結果が最終頁に与えられている。

5. タンデム待ち行列システムの最適制御

下図に示されるC段直列型M/M待ち行列システムを考える。客は平均到着率λのポアソン過程に従って到着し，i段の行列長はN_i $(i=1,\cdots,C)$に制限されているものとする。問題はi段のサーバー（サーバーiとよぶ）のとりうるサービス率が

$$\{0, \mu_i, \cdots, M_i\mu_i\} \qquad (i=1,\cdots,C)$$

で与えられる時，システムの状態に応じて各サーバーのサービス率をどのように制御すればよいかを決定することである。



システムの状態は，§2と同様客の到着あるいはサービスの終了時点で変化する。以下，§2の仮定(C1)－(C3)を仮定し，同じ費用構造を採用する。まず，客の到着あるいはサービスの終了直後におけるシステムの状態 S を

(5，1)　　$S=(X_1,\cdots,X_c,\ Y_1,\cdots,Y_c,\ Z_1,\cdots,Z_c)$

で表わす。ここで

$X_i\in\{0,1,\cdots,N_i\}$：i段の待ち行列長　$(i=1,\cdots,c)$

$Y_i\in\{0,1,\cdots,M_i\}$：(サーバーiの現在のサービス率)/μ_i

$$Z_i=\begin{cases}0, & \text{サーバー}\,i\,\text{が遊休あるいは休止中}\\ 1, & \text{〃　　稼動中}\qquad\qquad(i=1,\cdots,c-1)\\ 2, & \text{〃　　ブロックされている。}\end{cases}$$

$$Z_c=\begin{cases}0, & \text{サーバー}\,c\,\text{が遊休あるいは休止中}\\ 1, & \text{〃　　稼動中}\end{cases}$$

である。状態 S における決定 a は

(5．2)　　$a=(Y_1',\cdots,Y_c')$

で表わされる。ここで

$Y_i'\in\{0,1,\cdots,M_i\}$：(サーバー$i$のサービス率)/$\mu_i$

である。ブロックされているサーバーのサービス率も変えられるものとすれば，S でとりうる決定の集合 $A(S)$ は，

$Z_i=0$ or 2 $(i=1,\cdots,c-1)$，$Z_c=0$

となる各サーバーiについて Y_i' を $0\sim M_i$ まで変化させてえ

られる全ての組み合わせからなる。

状態 $S=(X_1,\cdots,X_c,\ Y_1,\cdots,Y_c,\ z_1,\cdots,z_c)$ で決定 $a=(Y_1',\cdots,Y_c')$ をとると，§2と同様中間状態

(5，3) $\quad S'=(X_1',\cdots,X_c',\ Y_1',\cdots,Y_c',\ z_1',\cdots,z_c')$

へ遷移する。ここで Y_i' は決定 a から定まる次の遷移までのサーバー i のサービス率を示し，$X_i',\ z_i'$ は次式から定められる。

(5，4) $$z_c'=\begin{cases}\varepsilon(X_c), & Y_c'\neq 0,\ z_c=0 \text{ のとき}\\ z_c, & \text{そうでないとき}\end{cases}$$

(5，5) $$X_c'=\begin{cases}X_c-\varepsilon(X_c), & Y_c'\neq 0,\ z_c=0,\ z_{c-1}\neq 2 \text{ のとき}\\ X_c, & \text{そうでないとき}\end{cases}$$

(5，6) $$z_c''=\begin{cases}0, & Y_c'\neq 0,\ z_c=0,\ z_{c-1}=2 \text{ のとき}\\ z_{c-1}, & \text{そうでないとき}\end{cases}$$

であり，$i=c-1,\cdots,1$ にたいして

(5，7) $$z_i'=\begin{cases}\varepsilon(X_i), & Y_i'\neq 0,\ z_{i+1}''=0 \text{ のとき}\\ z_{i+1}'', & \text{そうでないとき}\end{cases}$$

(5，8) $$X_i'=\begin{cases}X_i-\varepsilon(X_i), & Y_i'\neq 0,\ z_{i+1}''=0,\ z_{i-1}\neq 2 \text{ のとき}\\ & (z_0 \triangleq 0)\\ X_i, & \text{そうでないとき}\end{cases}$$

(5，9) $$z_i''=\begin{cases}0, & Y_i'\neq 0,\ z_{i+1}''=0,\ z_{i-1}=2 \text{ のとき}\\ z_{i-1}, & \text{そうでないとき} \quad (i\geq 2)\end{cases}$$

である。ここで $\varepsilon(X)$ は (2.11) で与えられている。(2.8) 同様，$i=1,\cdots,c$ にたいして

(5.10) $\quad \delta_i(X) = \begin{cases} 0 & , X = N_i のとき \\ 1 & , そうでないとき \end{cases}$

を定義し，

(5.11) $\quad \phi_j(z) = \begin{cases} 1 & , z = j \text{ のとき} \\ 0 & , そうでないとき \end{cases} \quad (j=1,2)$

とおく。中間状態 S で客が到着すればその直後に

(5.12) $\quad S_0 = (X_1'+\delta_1(X_1'), X_2', \cdots, X_c', Y_1', \cdots, Y_c', z_1', \cdots, z_c')$

へ遷移し，サーバー i のサービスが終了すればその直後に

(5.13) $\quad S_i = (X_1', \cdots, X_i', X_{i+1}'+\delta_{i+1}(X_{i+1}'), \cdots, X_c', Y_1', \cdots, Y_c', z_1', \cdots, z_{i-1}', 2(1-\delta_{i+1}(X_{i+1}')), z_{i+1}', \cdots, z_c') \quad (i<c)$

(5.14) $\quad S_c = (X_1', \cdots, X_c', Y_1', \cdots, Y_c', z_1', \cdots, z_{c-1}', 0)$

へ遷移する。割引遷移確率は §2 同様

(5.15) $\quad q_{SS_0}(a) = \lambda\delta_1(X_1') / \{\lambda\delta_1(X_1') + \sum_{i=1}^{c} Y_i'\mu_i\phi_1(z_i') + \alpha\}$

であり，$i=1,\cdots,c$ にたいして

(5.16) $\quad q_{SS_i}(a) = Y_i'\mu_i\phi_1(z_i') / \{\lambda\delta_1(X_1') + \sum_{i=1}^{c} Y_i'\mu_i\phi_1(z_i') + \alpha\}$

である．ブロックされているサーバーには遊休費用 w_{2i} が

かかるものとし，客にも待ち費用h_1がかかるものとすれば，平均割引費用$r(s, a)$は

$$(5.17)\qquad r(s,a)=\sum_{i=1}^{c}\{\mu_i d_i |Y_i - Y_i'| + e_i \varepsilon (Y_i - Y_i')\}$$
$$+\frac{1}{\{\lambda\delta_1(X_1') + \sum_{i=1}^{c} Y_i'\mu_i\phi_1(z_i') + \alpha\}}\times\{h_1(\sum_{i=1}^{c} X_i' + \sum_{i=1}^{c}\phi_2(z_i'))$$
$$+ h_2\sum_{i=1}^{c}\phi_1(z_i') + \sum_{i=1}^{c}\mu_i Y_i'(w_{1i}\phi_1(z_i') + w_{2i}(1-\phi_1(z_i')))\}$$

で与えられる。

6. 数値例

$c=2$, $N_1=10$, $N_2=2$, $\lambda=0.03$ $\mu_1=\mu_2=0.02$
$M_1=M_2=3$, $\alpha=0.01$, $h_1=h_2=10$, $w_{11}=w_{12}=250$
$w_{21}=w_{22}=125$, $d_1=d_2=5000$, $e_1=e_2=0$

を§3の修正政策反復法を用いて解き，唯一の最適定常政策（ステップ5）が58秒でえられた。ただし，$m=10$, $\varepsilon=0.01$とした。この問題の状態数は1588，とりうる決定の総数は11530であり，他の手法ではほぼ不可能な規模である。

7. おわりに

§4，6で$M/M/2$，$M/M\to M$ 待ち行列システムのサービス率の最適定常制御政策が§3の修正政策反復法で比較的容易にえられることが示された。今後より一層アルゴリ

ズムを改良するとともに実行しやすい準最適政策の性質を見い出し，その性質を上手く利用したアルゴリズムを開発することが残された課題である．

最後に常日頃御指導いただく，三根久教授に深謝するとともに全ての数値計算を実行していただいた市木邦美君に感謝いたします。

参考文献

[1] T.B. Crabill, D. Gross and M.J. Magazine, "A survey of research on optimal design and control of queues," Tech. paper T-280. The George Washington Univ. 1973.

[2] T.B. Crabill, D. Gross and M.J. Magazine, "A classified bibliography of research on optimal design and control of queues," Operations Res., 25, 219-232 (1977).

[3] N. Prabhu and S. Stidham,Jr. "Optimal control of queueing systems," in "Mathematical methods in queueing theory," pp.263-294. Springer, 1974.

[4] M.J. Sobel, "Optimal operation of queues," ibid. pp.231-261, Springer, 1974.

[5] 大野勝久，"待ち行列システムの最適制御" 数理解析研究所講究録 No.425, pp.106-127, 1981

[6] C.E. Bell, "Characterization and computation of optimal policies for operating an M/G/1 queueing system with removable servers," Operations Res., 19. 208-218 (1971).

[7] H.C. Tijms, "An algorithm for average costs denumerable state semi-Markov decision problems with applications to controlled production and decision

processes," in "Recent development in Markov decision processes," ed. by R. Hartley et al. pp.143-179. Academic. 1980.

[8] K. Ohno, "A unified approach to algorithms with a suboptimality test in discounted semi-Markov Decision Processes," J. Oper. Res. Soc. Japan, 24, 296-324 (1981).

[9] B.L. Fox and D.M. Landi, "An algorithm for identifying the ergodic subchains and transient states of a stochastic matrix," Comm. ACM 11, 619-621 (1968).

```
 LAM=    0.06000 MU=    0.01000 AL=    0.01000 EP=    0.01000
h1 HC=  10.00000 HS=    0.0      W1= 500.00000 W2= 250.00000 SC=5000.00000 SF=    0.0    m=10
 NUMBER=    0      h2          360MSEC                                   d              e
         1263980.89851
 NUMBER=    1                 1206MSEC
          314595.94907
 NUMBER=    2                 1219MSEC
           30699.60082
 NUMBER=    3                 1226MSEC
           20923.63452
 NUMBER=    4                 1264MSEC
           15682.92022
 NUMBER=    5                 1220MSEC
            3202.40208
 NUMBER=    6                 1201MSEC
            2353.45927
 NUMBER=    7                 1252MSEC
             619.97090
 NUMBER=    8                 1284MSEC
             224.00045
 NUMBER=    9                 1282MSEC
              82.50336
 NUMBER=   10                 1192MSEC
              36.61056
 NUMBER=   11                 1100MSEC
              16.21434
 NUMBER=   12                 1074MSEC
               7.23058
 NUMBER=   13                 1052MSEC
               3.26068
 NUMBER=   14                 1041MSEC
               1.44431
 NUMBER=   15                 1056MSEC
               0.63842
 NUMBER=   16                 1029MSEC
 CONVERGENCE                   297MSEC

 NUMBER OF ITERATIONS    17           UNIQUE-OPTIMAL POLICY IS OBTAINED.       CPU=   19355MSEC
   1000 1000   3225.818
   1100 1100   3264.053
   1110 1110   3727.362
   2000 2000   3216.675
   2100 2100   3256.910
   2101 2101   3720.219
   2110 2110   3674.732
   2200 2200   3295.146
   2210 2210   3677.673
   3000 3000   3238.052
   3100 3100   3280.360
   3101 3101   3716.900
   3110 3110   3660.423
                            3200 3200   3322.667
                            3201 3201   3676.720
                            3210 3210   3668.115
                            3300 3200   3372.667
                            3310 3310   3698.888
   4000 3000   3288.052
   4100 3100   3330.360
   4101 4101   3730.380
   4110 4110   3687.311
                            4200 3200   3372.667
                            4201 4201   3714.924
                            4210 4210   3698.150
```